Liver Risk Analysis

Sercan Tomaz

2024-06-29

First, let's load our dataset.

```
df <- read.csv("liver.risk.data.csv", sep = ",")
```

Let's examine the dimensions of our dataset.

```
dim(df)
```

```
## [1] 615  14
```

Let's move our response column to the last column.

```
library(dplyr)
df <- select(df, c(1, (3:14), 2))
```

The structure of our dataset

```
str(df)
```

```
## 'data.frame':    615 obs. of  14 variables:
##  $ X       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age     : int  32 32 32 32 32 32 32 32 32 32 ...
##  $ Sex     : chr  "m" "m" "m" "m" ...
##  $ ALB     : num  38.5 38.5 46.9 43.2 39.2 41.6 46.3 42.2 50.9 42.4 ...
##  $ ALP     : num  52.5 70.3 74.7 52 74.1 43.3 41.3 41.9 65.5 86.3 ...
##  $ ALT     : num  7.7 18 36.2 30.6 32.6 18.5 17.5 35.8 23.2 20.3 ...
##  $ AST     : num  22.1 24.7 52.6 22.6 24.8 19.7 17.8 31.1 21.2 20 ...
##  $ BIL     : num  7.5 3.9 6.1 18.9 9.6 12.3 8.5 16.1 6.9 35.2 ...
##  $ CHE     : num  6.93 11.17 8.84 7.33 9.15 ...
##  $ CHOL    : num  3.23 4.8 5.2 4.74 4.32 6.05 4.79 4.6 4.1 4.45 ...
##  $ CREA    : num  106 74 86 80 76 111 70 109 83 81 ...
##  $ GGT     : num  12.1 15.6 33.2 33.8 29.9 91 16.9 21.5 13.7 15.9 ...
##  $ PROT    : num  69 76.5 79.3 75.7 68.7 74 74.5 67.1 71.3 69.9 ...
##  $ Category: chr  "0=Blood Donor" "0=Blood Donor" "0=Blood Donor" "0=Blood Donor" ...
```

Let's now assign this "Sex" column as a factor.

```r
df$Sex <- factor(df$Sex, levels = c("m", "f"),labels = c("Male", "Female"))
str(df)
```

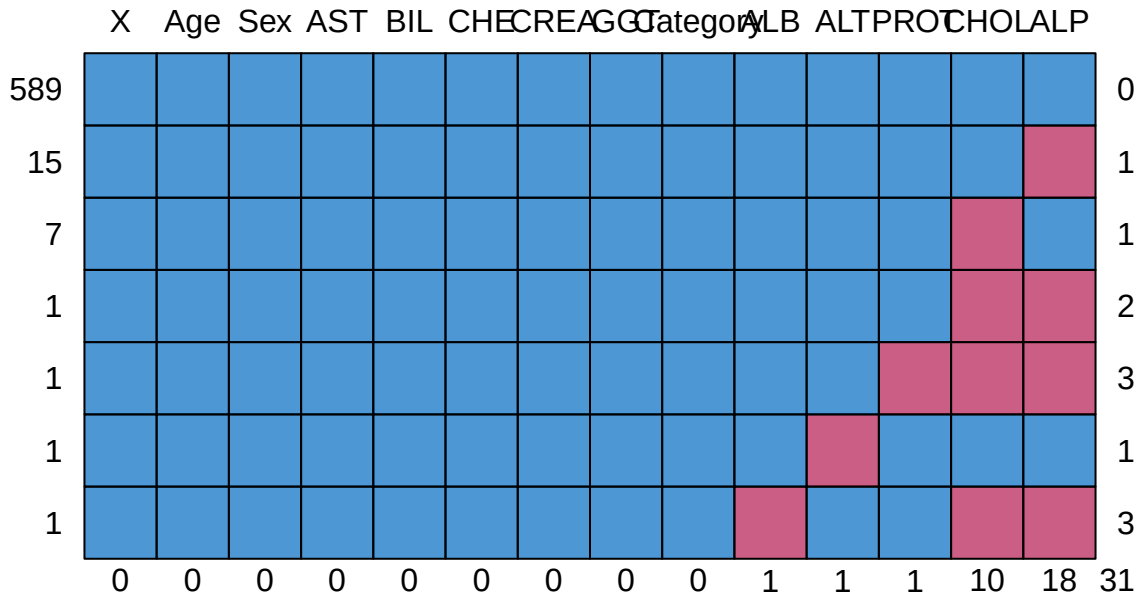
```
## 'data.frame':    615 obs. of  14 variables:
##  $ X       : int  1 2 3 4 5 6 7 8 9 10 ...
##  $ Age     : int  32 32 32 32 32 32 32 32 32 32 ...
##  $ Sex     : Factor w/ 2 levels "Male","Female": 1 1 1 1 1 1 1 1 1 1 ...
##  $ ALB     : num  38.5 38.5 46.9 43.2 39.2 41.6 46.3 42.2 50.9 42.4 ...
##  $ ALP     : num  52.5 70.3 74.7 52 74.1 43.3 41.3 41.9 65.5 86.3 ...
##  $ ALT     : num  7.7 18 36.2 30.6 32.6 18.5 17.5 35.8 23.2 20.3 ...
##  $ AST     : num  22.1 24.7 52.6 22.6 24.8 19.7 17.8 31.1 21.2 20 ...
##  $ BIL     : num  7.5 3.9 6.1 18.9 9.6 12.3 8.5 16.1 6.9 35.2 ...
##  $ CHE     : num  6.93 11.17 8.84 7.33 9.15 ...
##  $ CHOL    : num  3.23 4.8 5.2 4.74 4.32 6.05 4.79 4.6 4.1 4.45 ...
##  $ CREA    : num  106 74 86 80 76 111 70 109 83 81 ...
##  $ GGT     : num  12.1 15.6 33.2 33.8 29.9 91 16.9 21.5 13.7 15.9 ...
##  $ PROT    : num  69 76.5 79.3 75.7 68.7 74 74.5 67.1 71.3 69.9 ...
##  $ Category: chr  "0=Blood Donor" "0=Blood Donor" "0=Blood Donor" "0=Blood Donor" ...
```

Let's separate the "Category" column and reassign the required portion back to the "Category" column, and then delete the other column

```r
library(splitstackshape)
df <- cSplit(df, "Category", "=")
df <- select(df, -14)
colnames(df) [14] <- "Category"
```

Let's now identify the NA values and explore how we can replace them with appropriate values. The columns ALB, ALT, PROT, CHOL, and ALP contain NA values.

```r
library(mice)
md.pattern(df)
```




```
##     X Age Sex AST BIL CHE CREA GGT Category ALB ALT PROT CHOL ALP
## 589 1   1   1   1   1   1    1   1        1   1   1    1    1   0
## 15  1   1   1   1   1   1    1   1        1   1   1    1    1   1
## 7   1   1   1   1   1   1    1   1        1   1   1    1    0   1
## 1   1   1   1   1   1   1    1   1        1   1   1    1    0   2
## 1   1   1   1   1   1   1    1   1        1   1   1    0    0   3
## 1   1   1   1   1   1   1    1   1        1   1   0    1    1   1
## 1   1   1   1   1   1   1    1   1        1   0   1    1    0   3
##     0   0   0   0   0   0    0   0        0   1   1    1   10  18 31
```

For the ALB column, we need to perform missing value imputation.

```
av.alb <- df %>%
  group_by(Category) %>%
  summarise(mean.alb = mean(ALB, na.rm = TRUE))

for (i in 1:nrow(df)) {
  if (is.na(df$ALB[i])) {
    df$ALB[i] <- av.alb$mean.alb[av.alb$Category == df$Category[i]]
  }
}
```

For the ALT column, missing value imputation has been done.

```
avg.alt <- df %>%
  group_by(Category) %>%
  summarise(mean.alt = mean(ALT, na.rm = TRUE))

for (i in 1:nrow(df)) {
  if (is.na(df$ALT [i])) {
    df$ALT[i] <- avg.alt$mean.alt [avg.alt$Category == df$Category [i]]
  }
}
```

For the PROT column, missing value imputation has been done.

```
avg.prot <- df %>%
  group_by(Category) %>%
  summarize(mean.prot = mean(PROT, na.rm = TRUE))

for (i in 1:nrow(df)) {
  if (is.na(df$PROT [i])) {
    df$PROT [i] <- avg.prot$mean.prot [avg.prot$Category == df$Category [i]]
  }
}
```

For the CHOL column, missing value imputation has been done.

```
avg.chol <- df %>%
  group_by(Category) %>%
  summarize(mean.chol = mean(CHOL, na.rm = TRUE))

for (i in 1:nrow(df)) {
  if (is.na(df$CHOL [i])) {
    df$CHOL [i] <- avg.chol$mean.chol [avg.chol$Category == df$Category [i]]
  }
}
```

For the ALP column, missing value imputation has been done.

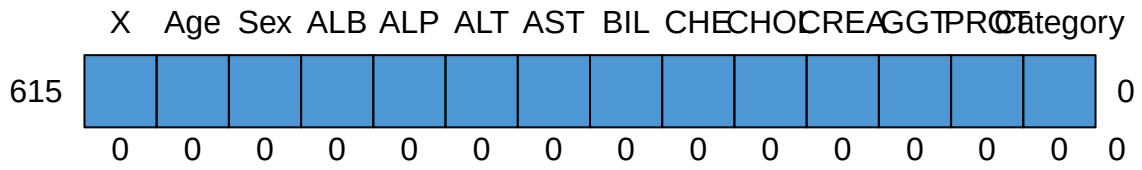
```
avg.alp <- df %>%
  group_by(Category) %>%
  summarize(mean.alp = mean(ALP, na.rm = TRUE))

for (i in 1:nrow(df)) {
  if (is.na(df$ALP [i])) {
    df$ALP [i] <- avg.alp$mean.alp [avg.alp$Category == df$Category [i]]
  }
}
```

We have completed our missing value imputation process, and we have replaced the missing values with the category-wise averages. Now, let's review and verify that we have obtained the complete dataset.

```
md.pattern(df)
```

```
##  /\     /\
```

```
## {  '---'  }
## {  O   O  }
## ==>  V <==  No need for mice. This data set is completely observed.
##  \  \|/  /
##   '-----'
```



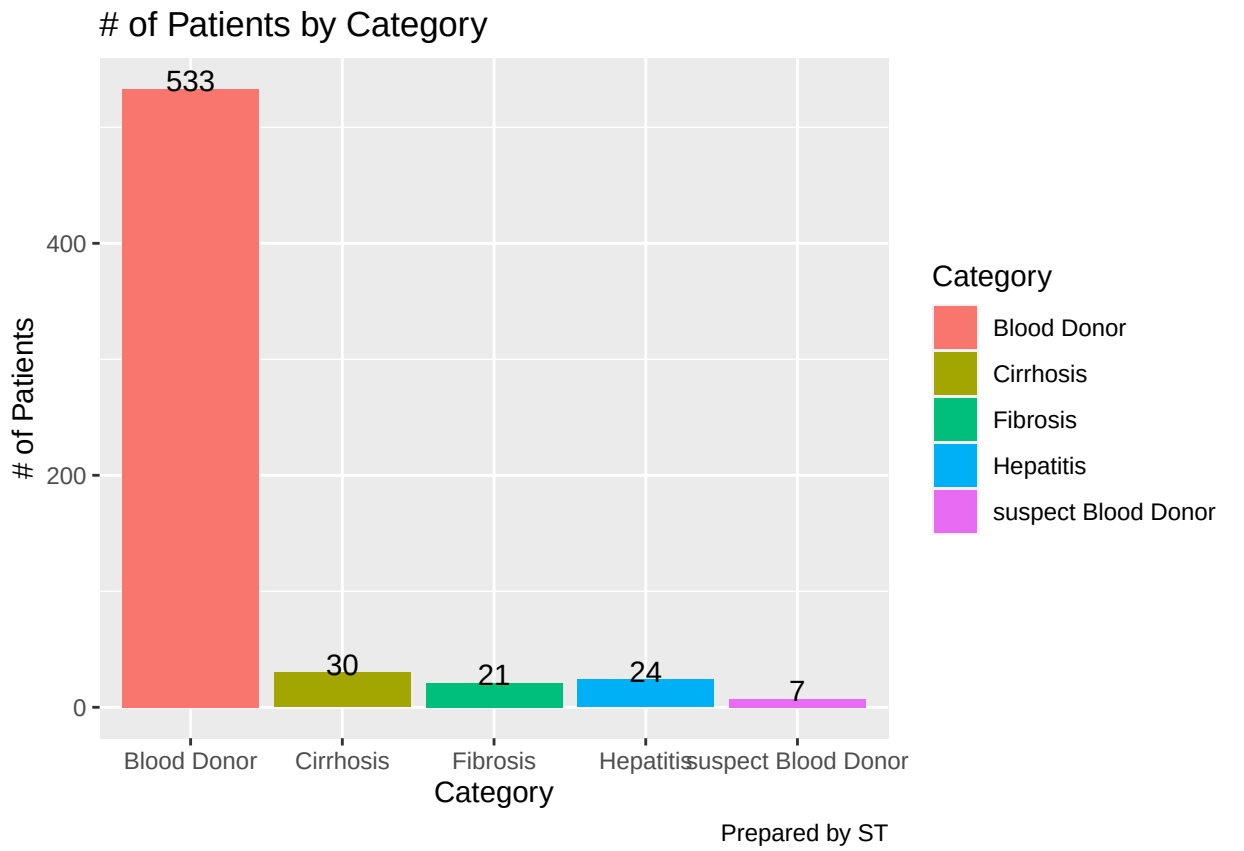

```
##     X Age Sex ALB ALP ALT AST BIL CHE CHOL CREA GGT PROT Category
## 615 1   1   1   1   1   1   1   1   1    1    1   1    1        1 0
##     0   0   0   0   0   0   0   0   0    0    0   0    0        0 0
```

Now, let's examine the relationships between the columns in our dataset through EDA.

First, let's examine the number of patients by category. Looking at our graph, we observe a statistically unbalanced distribution. The majority of patients fall under the Blood Donor category, and there is a noticeable difference when compared to the number of patients in other categories. This is an important factor to consider when developing machine learning models, as predictions for categories with fewer patients may be less accurate or some patients may be missed.

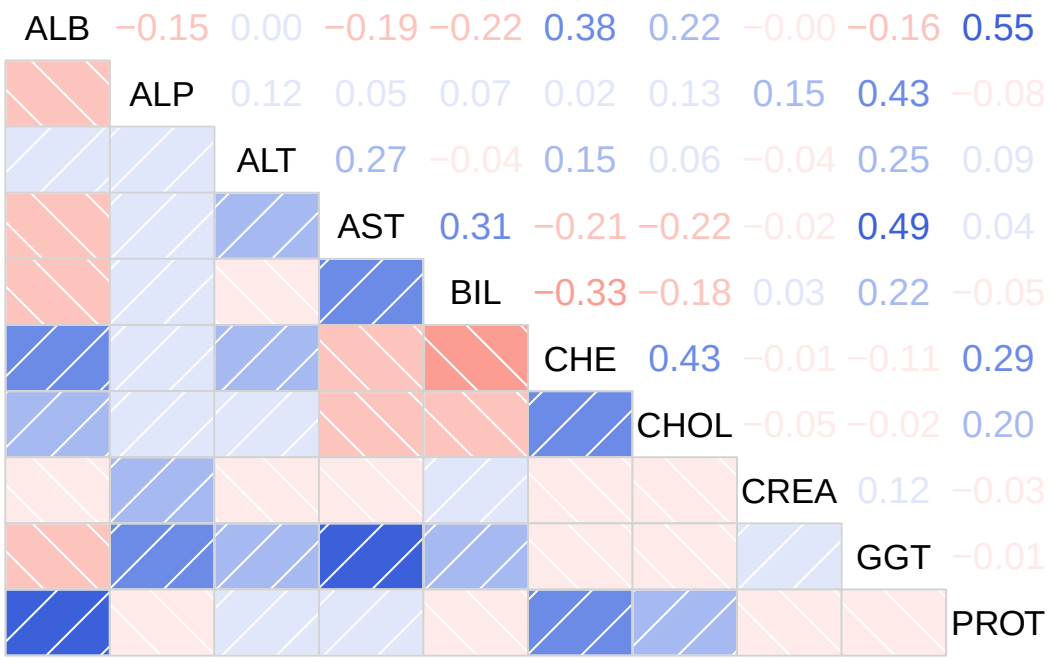
```
library(ggplot2)
ggplot(df, aes(x = Category)) +
  geom_bar(aes(fill = Category)) +
  ggtitle("# of Patients by Category") +
  labs(x = "Category",
       y = "# of Patients",
       caption = "Prepared by ST") +
```

```
  geom_text(stat = "count", aes(label = ..count..),
            position = position_dodge(width = 0.9),
            vjust = 0.1,
            hjust = 0.5)
```



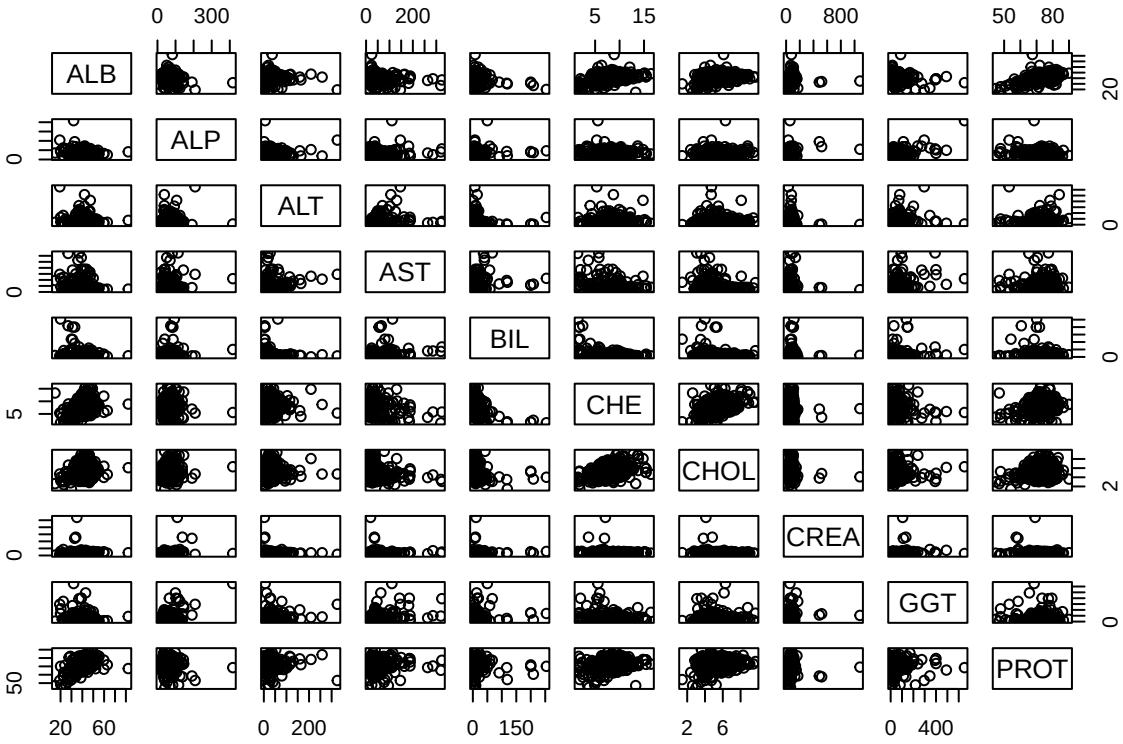
Based on the provided correlation matrix, several variables exhibit significant correlations. Albumin (ALB) and Protein (PROT) show a strong positive correlation. Aspartate Aminotransferase (AST) and Gamma-Glutamyl Transferase (GGT) demonstrate a moderate positive correlation, as do Cholinesterase (CHE) and Cholesterol (CHOL), as well as GGT and Alkaline Phosphatase (ALP). Additionally, CHE and PROT, CHE and ALB, and ALB and CHOL each display moderate positive correlations. These correlations should be considered in medical analyses and modeling.

```
library(corrgram)
corrgram(df [, 4:13], upper.panel = panel.cor)
```



We examined the correlation matrix and identified the variables that are correlated with each other. Now, let's visualize them collectively on a scatter plot.

```
pairs(df [, 4:13])
```



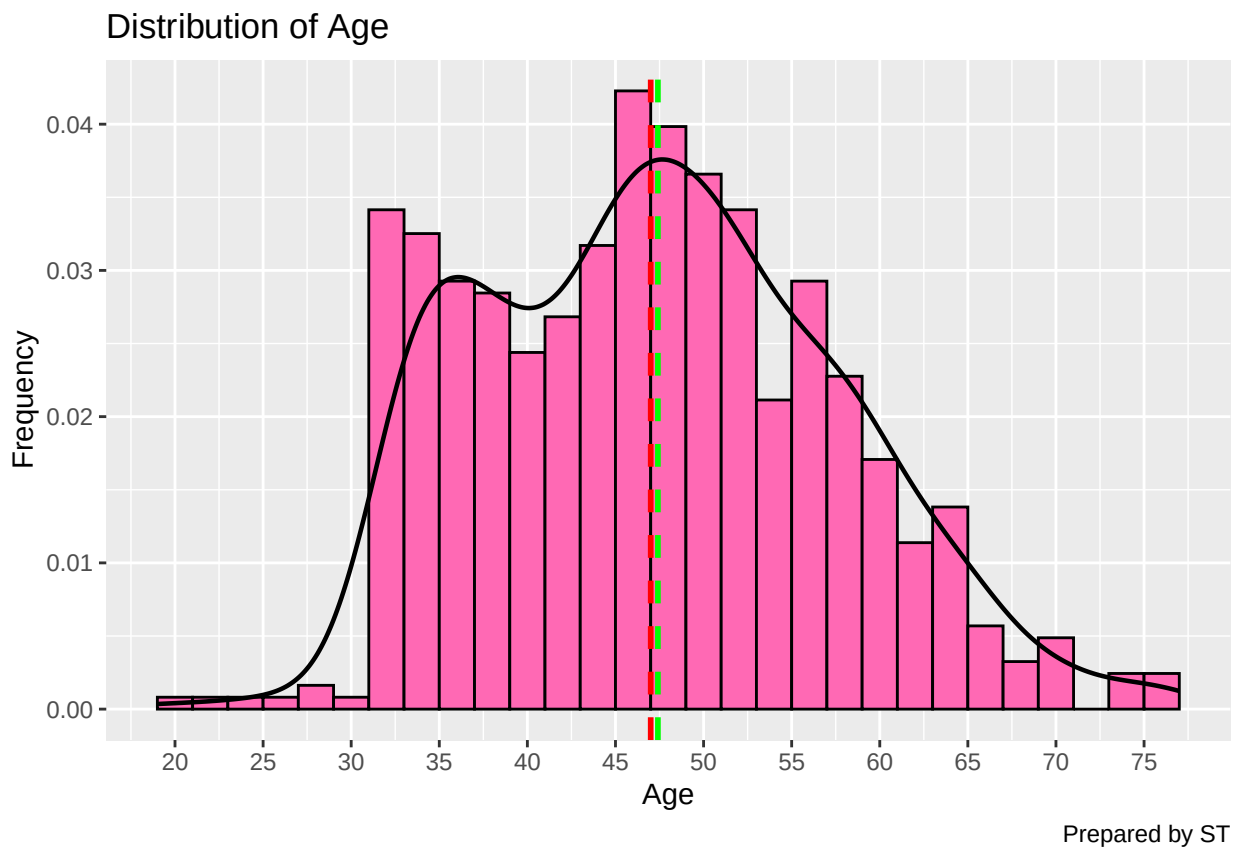
We are calculating the average age of the patients, and median in order to display it on a histogram of ages.

```
mean.age <- mean(df$Age)
median.age <- median(df$Age)
```

Upon examining our histogram, we can make the following observations. Our histogram is not skewed to either side, hence it can be considered to have an ALMOST SYMMETRIC DISTRIBUTION. The dashed line highlighted in green represents the mean, which is 47.41 years old. Meanwhile, our median, depicted in red, is very close to the mean age at 47. Based on this, we can reaffirm and demonstrate that our dataset is symmetric. Looking at the distribution of ages, we observe a concentration of patients around the mean age. Additionally, I notice a potential outlier at the age of 75, but this is merely an interpretation based on the histogram. The safest conclusion will be provided by a box plot analysis.

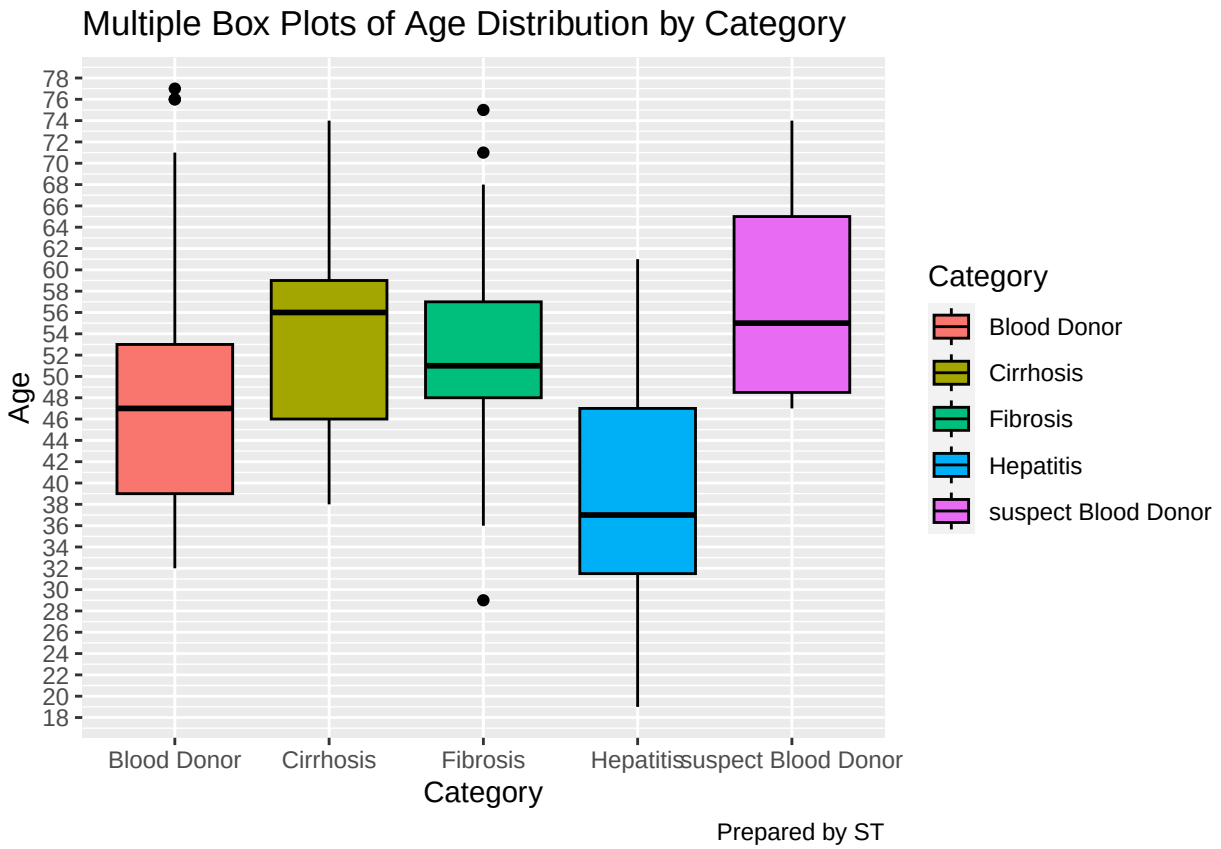
```
ggplot(df, aes(x = Age)) +
  geom_histogram(aes(y = ..density..),
                 bins = 30,
                 color = "black",
                 fill = "hotpink") +
  scale_x_continuous(breaks = seq(0, 80, by = 5)) +
  geom_density(color = "black",
               size = 0.8) +
  geom_vline(aes(xintercept = mean.age),
             linetype = "dashed",
             size = 1,
             color = "green") +
  ggtitle("Distribution of Age") +
```

```
  labs(x = "Age",
       y = "Frequency",
       caption = "Prepared by ST"
  ) +
  geom_vline(aes(xintercept = median.age),
             linetype = "dashed",
             size = 1,
             color = "red")
```



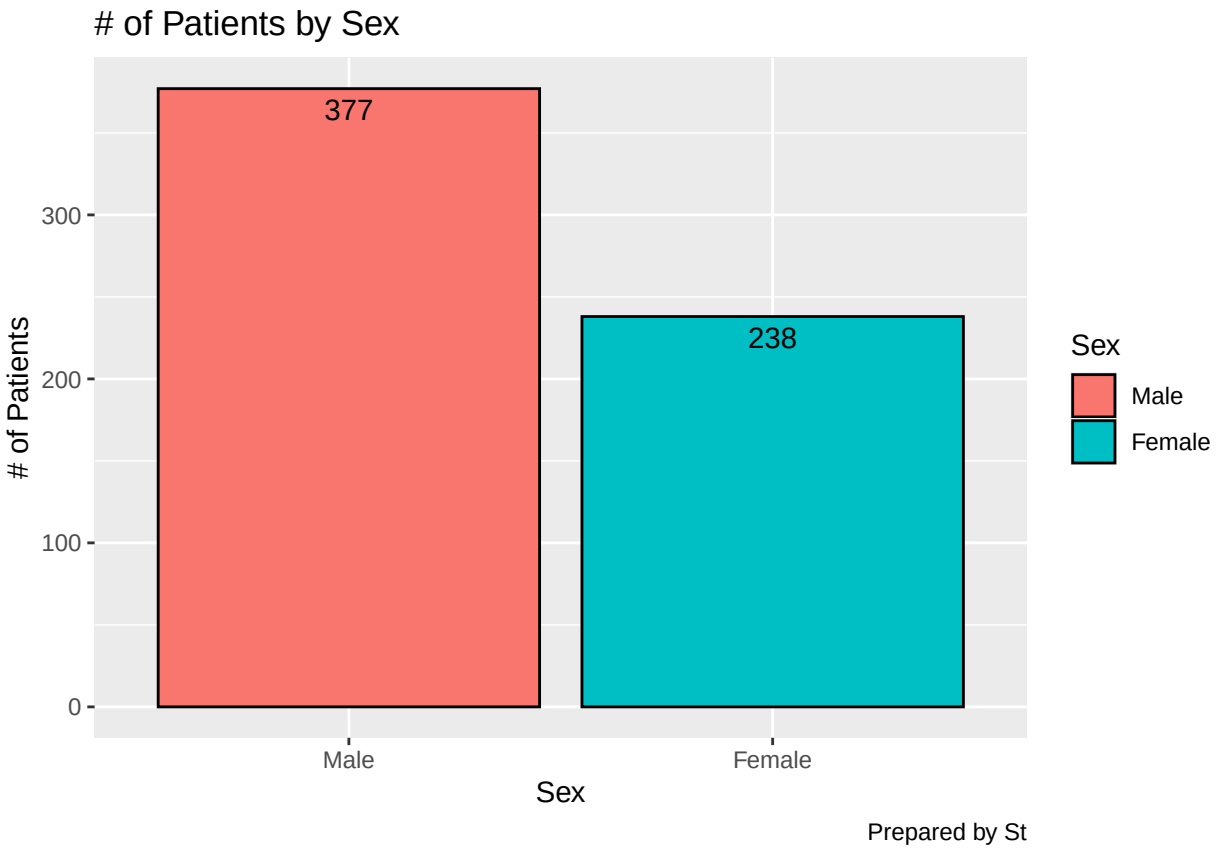
When examining the multiple box plots, we observe that there are outlier values within different age groups. The average age is highest for Cirrhosis patients, while the average age is lowest for Hepatitis patients.

```
ggplot(df, aes(x = Category,
               y = Age)) +
  geom_boxplot(aes(fill = Category),
               color = "black") +
  scale_y_continuous(breaks = seq(18, 80, by = 2)) +
  ggtitle("Multiple Box Plots of Age Distribution by Category") +
  labs(caption = "Prepared by ST")
```



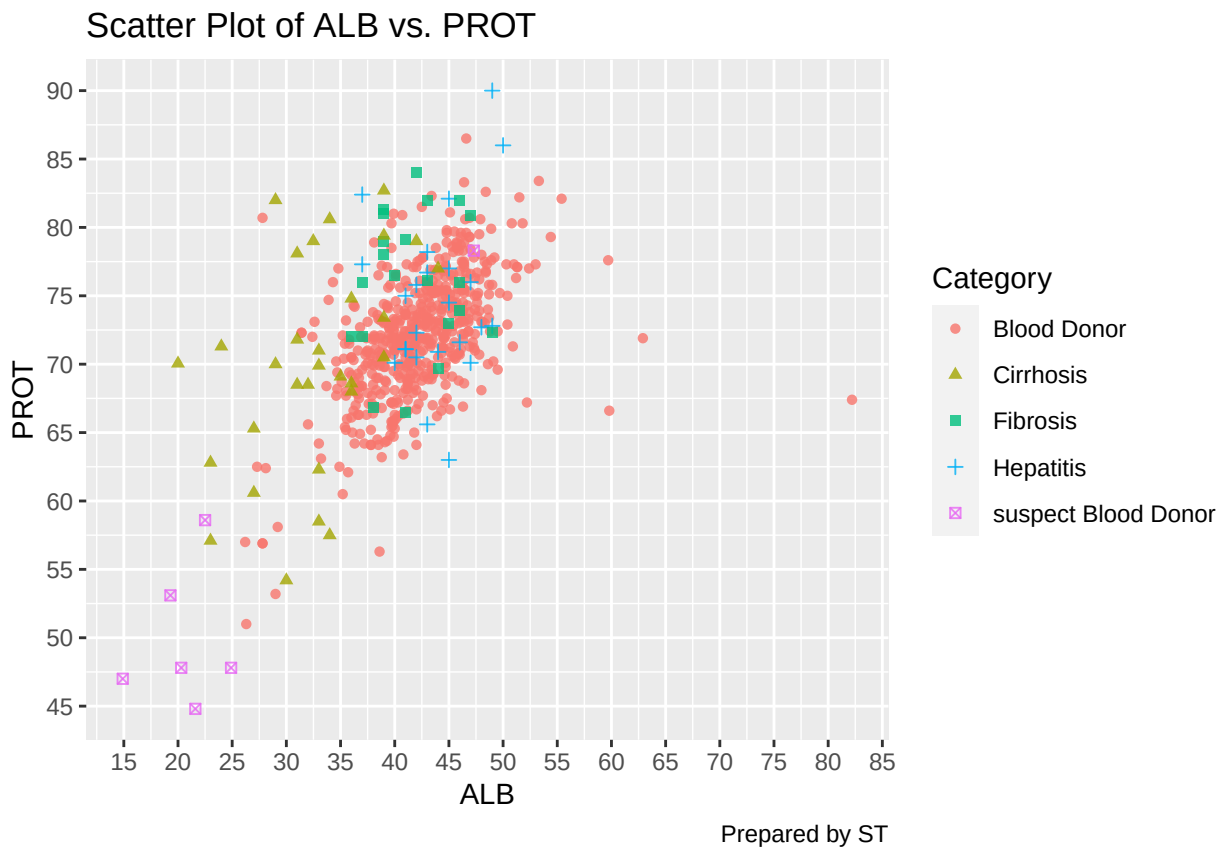
When examining the bar plot for genders, we see that males are the majority with 377, while females number 238.

```
ggplot(df, aes(x = Sex)) +
  geom_bar(aes(fill = Sex),
           color = "black") +
  geom_text(stat = "count", aes(label = ..count..),
            position = position_dodge(width = 0.9),
            vjust = 1.5,
            hjust = 0.5) +
  ggtitle("# of Patients by Sex") +
  labs(y = "# of Patients",
       caption = "Prepared by St")
```



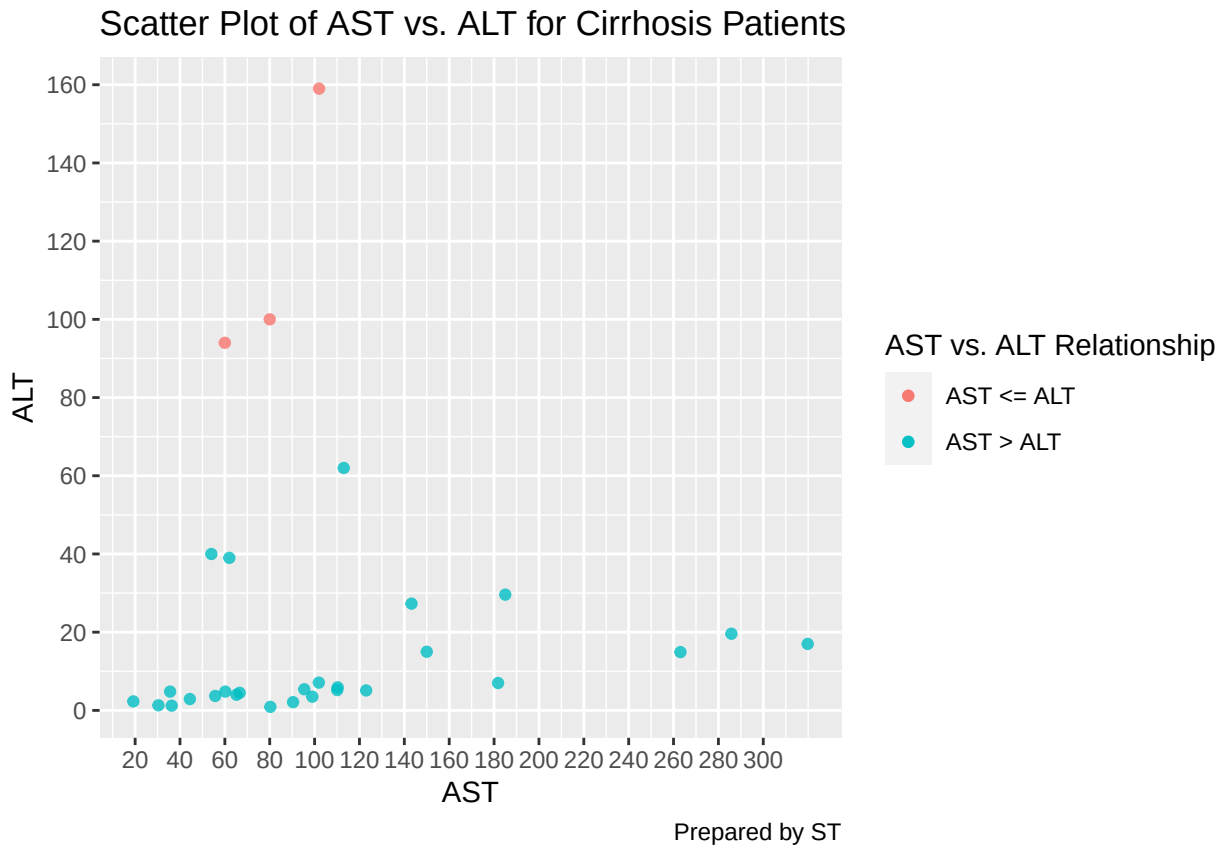
When we examine the ALB / PROT scatter plot, we observe a strong positive correlation of 0.55. Patients in the suspect blood donor category tend to have lower ALB and PROT values compared to those in other categories. Looking at the cirrhosis category, we particularly notice that the ALB value is lower compared to the fibrosis and hepatitis categories. However, fibrosis and hepatitis patients have higher ALB values compared to patients in the cirrhosis category.

```
ggplot(df, aes(x = ALB,
               y = PROT)) +
  geom_point(aes(color = Category,
                 shape = Category),
             size = 1.5,
             alpha = 0.8) +
  scale_x_continuous(breaks = seq(0, 90, by = 5)) +
  scale_y_continuous(breaks = seq(0, 90, by = 5)) +
  ggtitle("Scatter Plot of ALB vs. PROT") +
  labs(caption = "Prepared by ST")
```



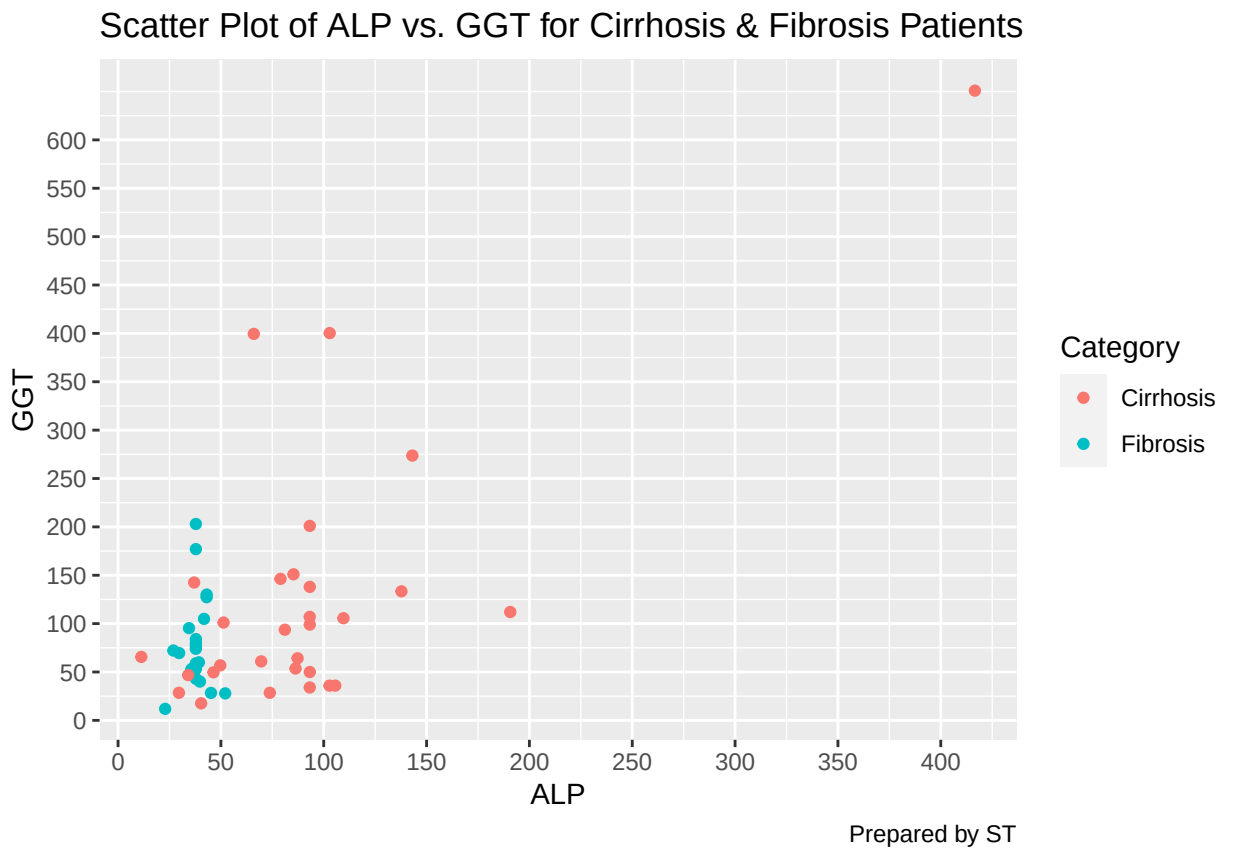
When examining the AST vs. ALT scatter plot for cirrhotic patients, the following observation can be made: Upon reviewing the graph, it is noteworthy that the predominance of AST > ALT condition is apparent among patients in the cirrhosis category. As non-medical professionals, we are making interpretations solely based on the values, and one of the significant interpretations we can make from this plot is the potential role of AST/ALT ratio in detecting liver diseases.

```
ggplot(data = subset(df, Category = "Cirrhosis"), aes(x = AST, y = ALT)) +
  geom_point(data = subset(df, AST > ALT & Category == "Cirrhosis"),
             size = 1.5, alpha = 0.8, aes(color = "AST > ALT")) +
  geom_point(data = subset(df, AST <= ALT & Category == "Cirrhosis"),
             size = 1.5, alpha = 0.8, aes(color = "AST <= ALT")) +
  scale_x_continuous(breaks = seq(0, 300, by = 20)) +
  scale_y_continuous(breaks = seq(0, 300, by = 20)) +
  ggtitle("Scatter Plot of AST vs. ALT for Cirrhosis Patients") +
  labs(caption = "Prepared by ST",
       color = "AST vs. ALT Relationship")
```



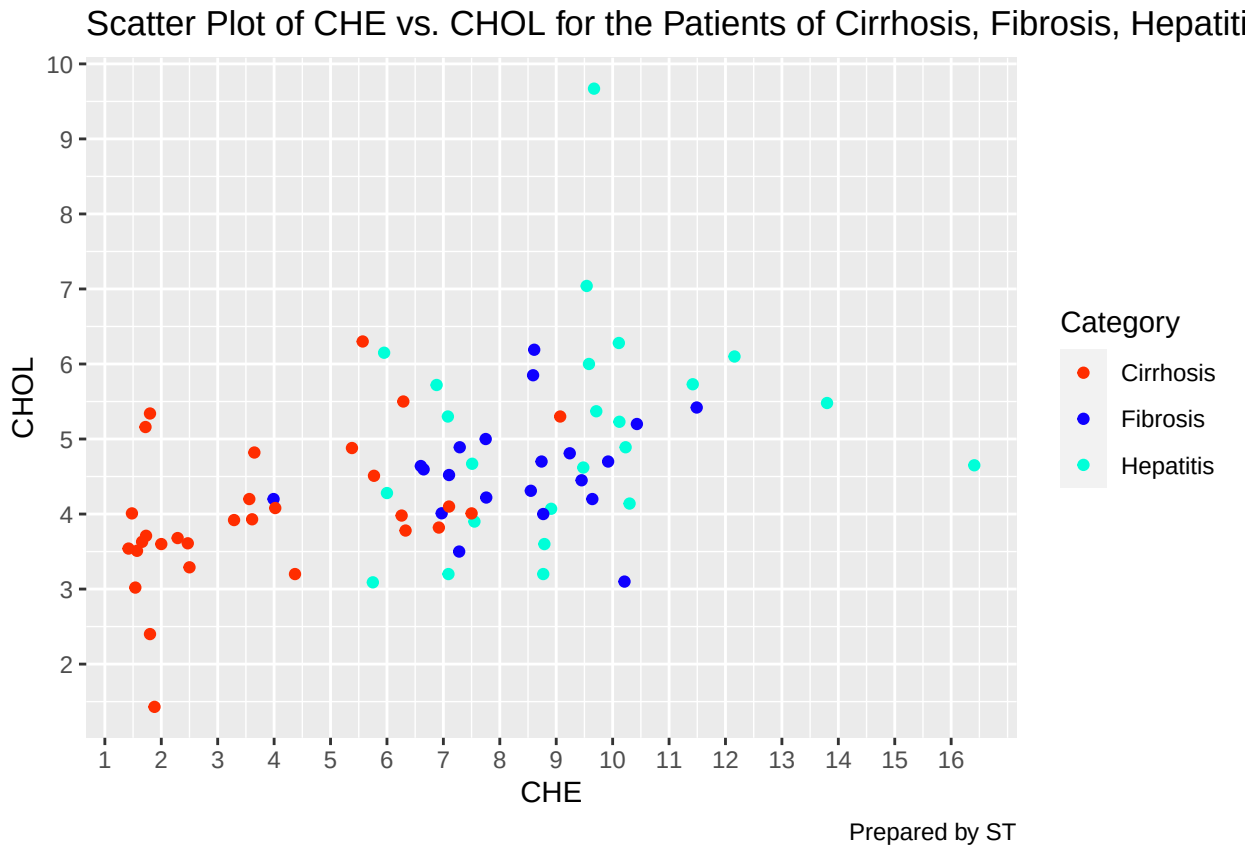
When we examine the ALP vs. GGT scatter plot for cirrhosis and fibrosis patients, we can make the following observations: Cirrhosis patients tend to have higher ALP values compared to fibrosis patients. At the same time, there is a positive correlation of 0.43 between them.

```
ggplot(data = subset(df, Category == "Cirrhosis" | Category == "Fibrosis"), aes(x = ALP, y = GGT)) +
  geom_point(aes(color = Category)) +
  scale_x_continuous(breaks = seq(0, 400, by = 50)) +
  scale_y_continuous(breaks = seq(0, 600, by = 50)) +
  ggtitle("Scatter Plot of ALP vs. GGT for Cirrhosis & Fibrosis Patients") +
  labs(caption = "Prepared by ST")
```



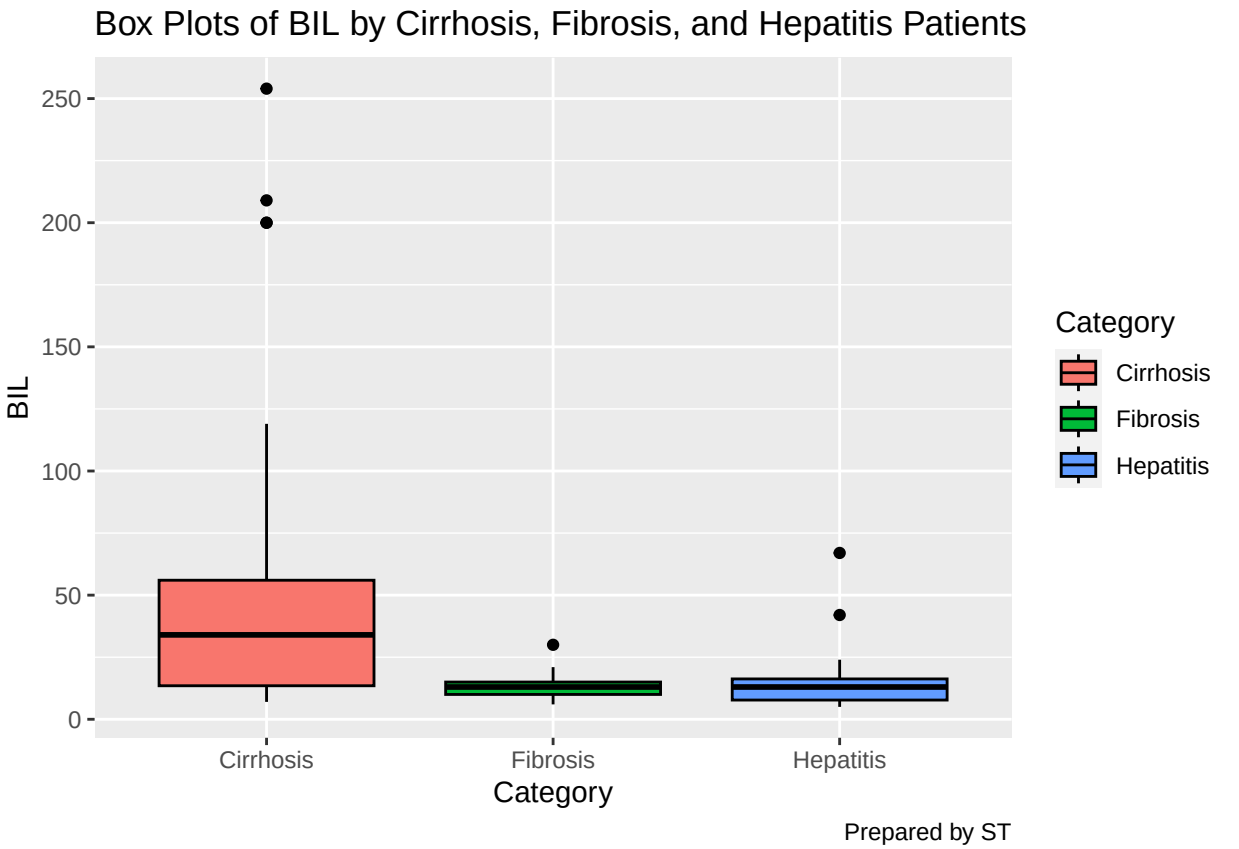
In examining the scatter plot for CHE (cholinesterase) versus CHOL (cholesterol) values among patients with Cirrhosis, Fibrosis, and Hepatitis, we gain valuable insights into the impact of these conditions on liver functions. Notably, it is evident that patients with Cirrhosis exhibit significantly lower CHE values compared to those with Fibrosis and Hepatitis. This observation suggests a more pronounced impairment of liver function in Cirrhosis patients. In contrast, for patients with Fibrosis and Hepatitis, it is challenging to draw similar conclusions due to the variability in CHE and CHOL values, which tend to fluctuate based on the severity of the disease. Generally, these values exhibit less pronounced changes compared to Cirrhosis patients and are more dispersed across the plot. In summary, the scatter plot reveals that CHE and CHOL values are generally lower in Cirrhosis patients, with data points predominantly clustered in the lower-left region of the plot. Conversely, while there may be reductions in CHE and CHOL values for Fibrosis and Hepatitis patients, these reductions are less pronounced and tend to vary according to the severity of the disease, resulting in a more dispersed distribution of data points across the plot.

```
ggplot(data = subset(df, Category == "Cirrhosis" |
                         Category == "Hepatitis" |
                         Category == "Fibrosis"),
       aes(x = CHE, y = CHOL)) +
  geom_point(aes(color = Category)) +
  scale_x_continuous(breaks = seq(0, 16, by = 1)) +
  scale_y_continuous(breaks = seq(0, 10, by = 1)) +
  labs(caption = "Prepared by ST") +
  ggtitle("Scatter Plot of CHE vs. CHOL for the Patients of Cirrhosis, Fibrosis, Hepatitis") +
  scale_color_manual(values = c("Cirrhosis" = "#FF2D00",
                                "Hepatitis" = "#00FFD8",
                                "Fibrosis" = "#0F00FF"))
```



Let's compare the BIL (bilirubin) values for patients with Cirrhosis, Fibrosis, and Hepatitis. Upon examination, it is evident that Cirrhosis patients have significantly higher average BIL values compared to the other two patient groups. When reviewing a graph for Fibrosis and Hepatitis patients, we observe that the average BIL values are nearly equal. In summary, Cirrhosis patients generally exhibit the highest average BIL values. This indicates severe impairment of liver function, resulting in inefficient processing of bilirubin by the liver. BIL values for Fibrosis and Hepatitis patients vary depending on the progression of the disease. Additionally, all three patient categories show outlier values in terms of BIL levels.

```
ggplot(data = subset(df, Category == "Cirrhosis" |
                       Category == "Hepatitis" |
                       Category == "Fibrosis"),
       aes(x = Category, y = BIL)) +
  geom_boxplot(aes(fill = Category), color = "black") +
  ggtitle("Box Plots of BIL by Cirrhosis, Fibrosis, and Hepatitis Patients") +
  labs(caption = "Prepared by ST")
```



We have thoroughly analyzed our dataset, examining the relationships between various variables and interpreting them by categories. Our next step involves dividing the dataset into training and testing sets. Subsequently, we will construct several machine learning models to select the most consistent model based on parameters. This approach aims to identify the model that best fits the data and optimizes performance. We will evaluate different machine learning algorithms to achieve this goal.

```
library(caTools)
set.seed(101)
df <- select(df, -X)
df$Category <- as.factor(df$Category)
split <- sample.split(df$Category, SplitRatio = 0.7)
train <- subset(df, split == TRUE)
test <- subset(df, split == FALSE)
```

```
library(caret)
library(rpart)
library(rpart.plot)
param.grid.ct <- expand.grid(cp = seq(0.01, 1, by = 0.01))
```

```
set.seed(101)
ctrl.ct <- trainControl(method = "cv",
                        number = 5)
```

```
set.seed(101)
parameter.search.ct <- train(Category ~.,
```

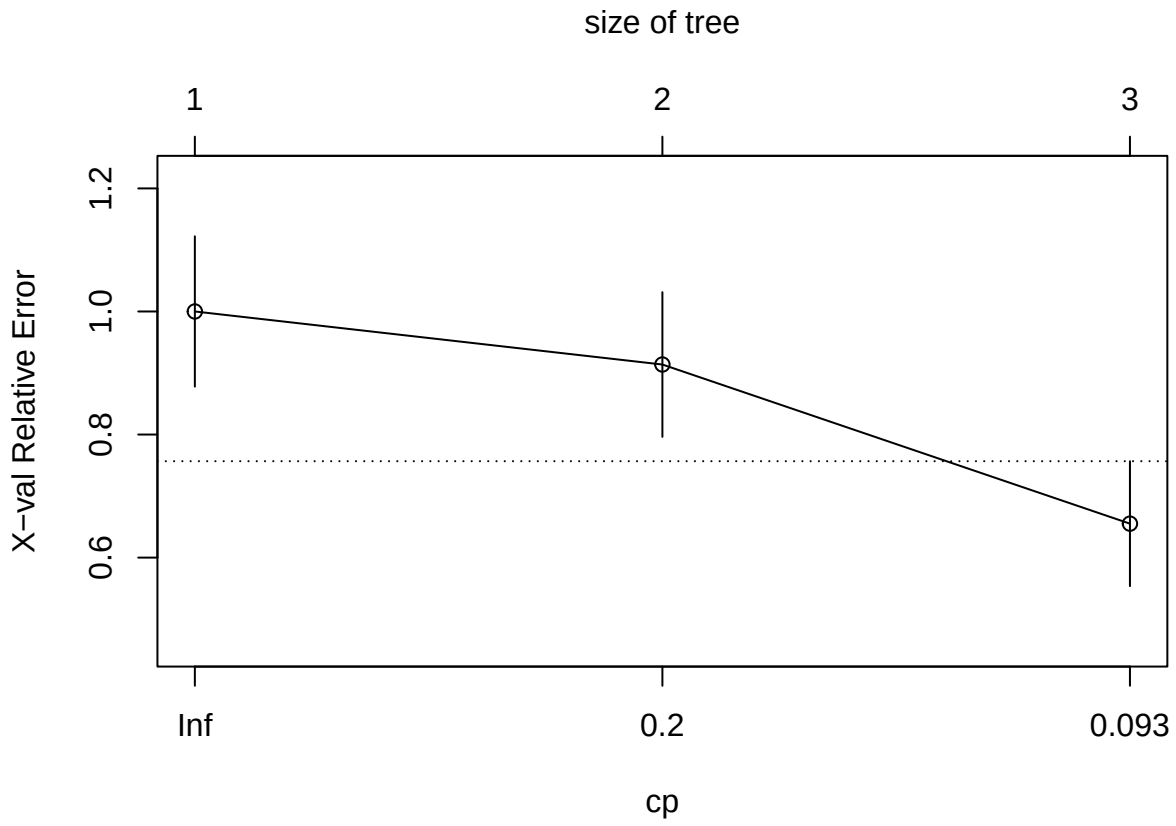
```
                              data = train,
                              method = "rpart",
                              trControl = ctrl.ct,
                              tuneGrid = param.grid.ct)
```

```
set.seed(101)
parameter.search.ct$bestTune$cp
```

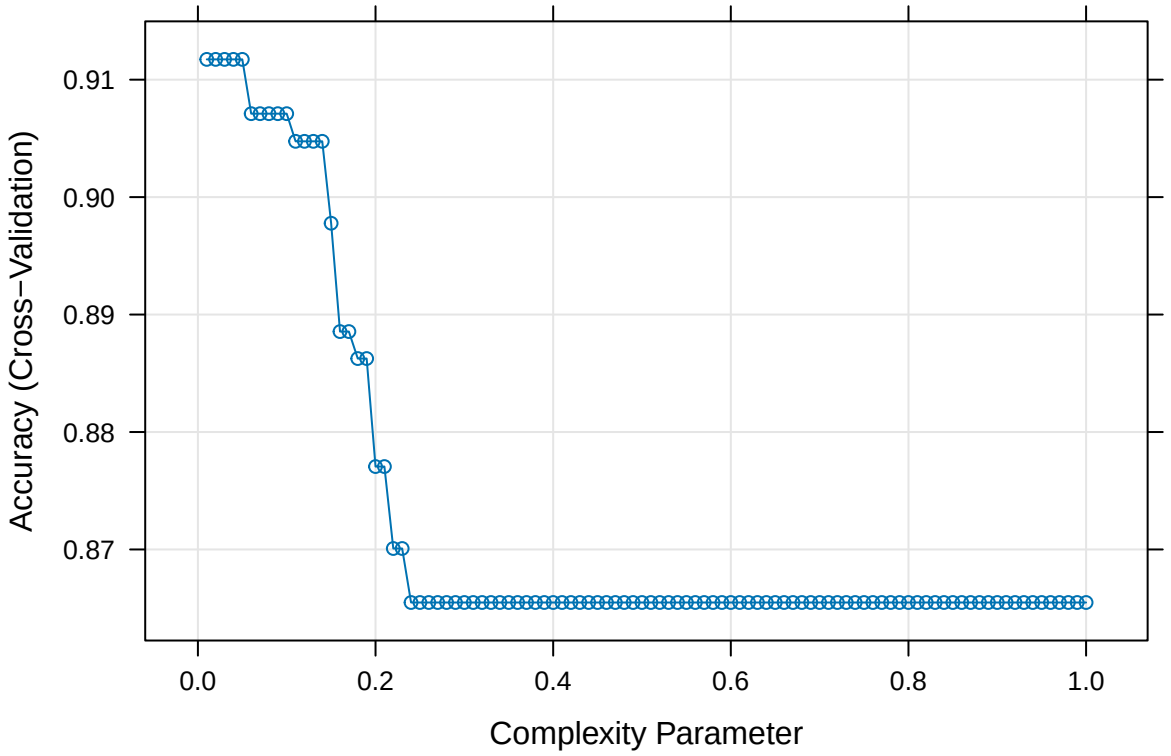
```
## [1] 0.05
```

```
set.seed(101)
ct.model <- rpart(Category ~.,
                  data = train,
                  cp = parameter.search.ct$bestTune$cp)
```

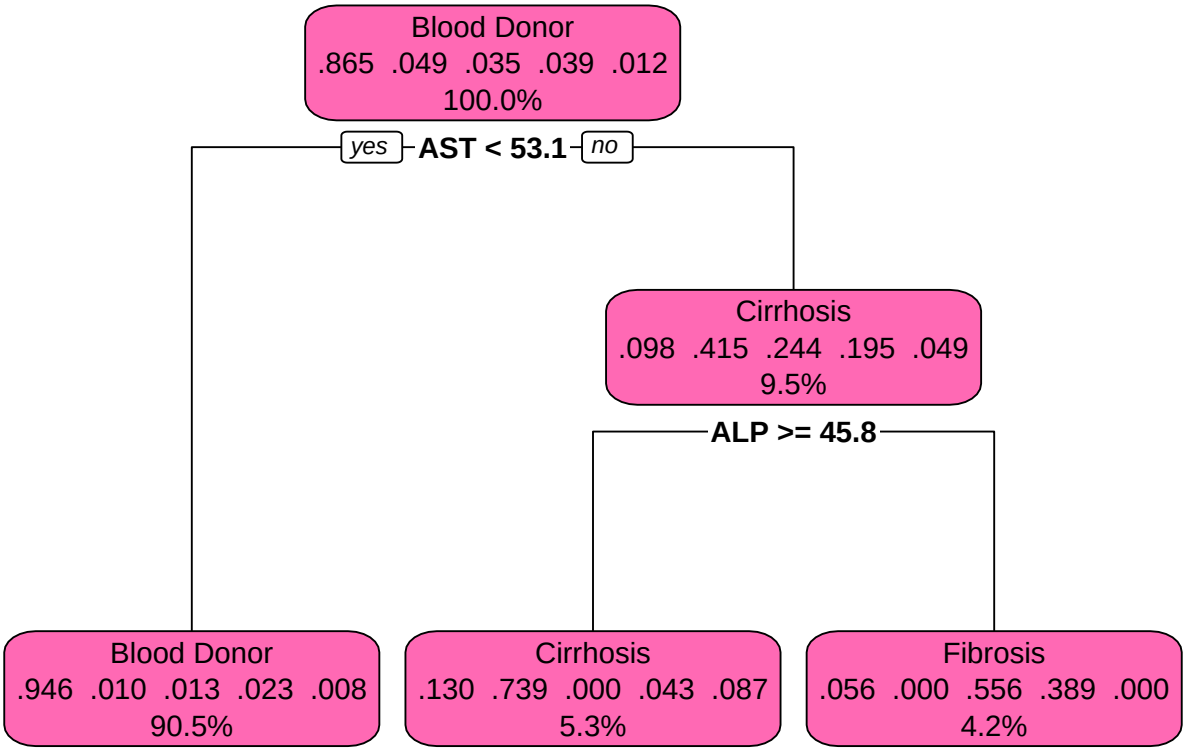
```
plotcp(ct.model)
```




```
plot(parameter.search.ct)
```




```
rpart.plot(ct.model, digits = 3, box.palette = "hotpink")
```




```
ct.preds <- predict(ct.model, test, type = "class")
```

Now, let us examine the performance metrics of the classification tree model we have developed. Starting with sensitivity, we observe high sensitivity values for the Blood Donor and Fibrosis categories. This indicates that our model effectively identifies blood donors and fibrosis patients. On the other hand, it struggles to detect Cirrhosis and Hepatitis & Suspect Blood Donor cases. When we consider specificity values, we see that the model accurately identifies individuals who do not have Cirrhosis, Fibrosis, and Hepatitis & Suspect Blood Donor conditions. However, it faces challenges in correctly identifying those who are not blood donors within the blood donor category. It is important to note that our dataset is statistically imbalanced, which can be inferred by looking at the prevalence. To improve the quality of the dataset, data collection efforts should focus on the groups with low prevalence.

```
confusionMatrix(ct.preds, test$Category)
```

```
## Confusion Matrix and Statistics
##
##                     Reference
## Prediction           Blood Donor Cirrhosis Fibrosis Hepatitis
##   Blood Donor                159         1        1         4
##   Cirrhosis                    1         4        0         1
##   Fibrosis                     0         4        5         2
##   Hepatitis                    0         0        0         0
##   suspect Blood Donor          0         0        0         0
##                     Reference
## Prediction           suspect Blood Donor
```

```
##   Blood Donor                          1
##   Cirrhosis                            1
##   Fibrosis                             0
##   Hepatitis                            0
##   suspect Blood Donor                  0
##
## Overall Statistics
##
##                Accuracy : 0.913
##                  95% CI : (0.8626, 0.9495)
##     No Information Rate : 0.8696
##     P-Value [Acc > NIR] : 0.04483
##
##                   Kappa : 0.5892
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Blood Donor Class: Cirrhosis Class: Fibrosis
## Sensitivity                      0.9938          0.44444         0.83333
## Specificity                      0.7083          0.98286         0.96629
## Pos Pred Value                   0.9578          0.57143         0.45455
## Neg Pred Value                   0.9444          0.97175         0.99422
## Prevalence                       0.8696          0.04891         0.03261
## Detection Rate                   0.8641          0.02174         0.02717
## Detection Prevalence             0.9022          0.03804         0.05978
## Balanced Accuracy                0.8510          0.71365         0.89981
##                      Class: Hepatitis Class: suspect Blood Donor
## Sensitivity                   0.00000                    0.00000
## Specificity                   1.00000                    1.00000
## Pos Pred Value                    NaN                        NaN
## Neg Pred Value                0.96196                    0.98913
## Prevalence                    0.03804                    0.01087
## Detection Rate                0.00000                    0.00000
## Detection Prevalence          0.00000                    0.00000
## Balanced Accuracy             0.50000                    0.50000
```

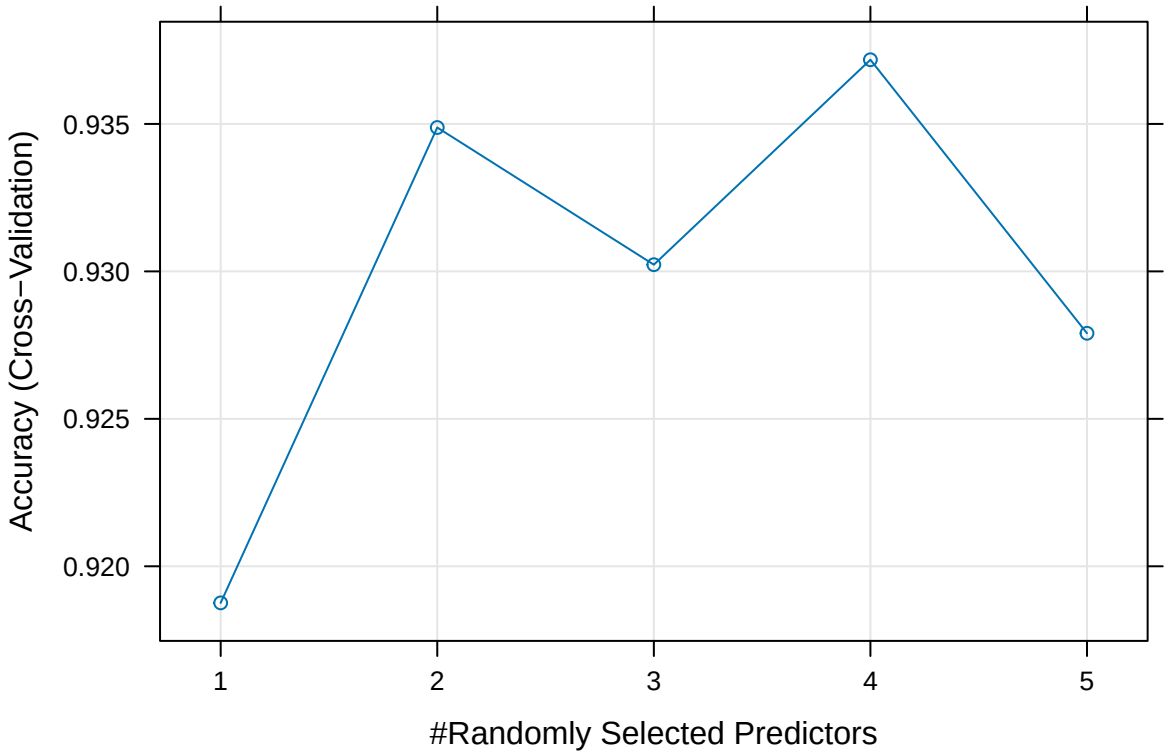
Now, let's develop our second model using a random forest algorithm, fine-tuning the necessary parameters.

```
param.grid.rf <- expand.grid(mtry = c(1, 2, 3, 4, 5))
```

```
set.seed(101)
ctrl.rf <- trainControl(method = "cv",
                        number = 5)
```

```
parameter.search.rf <- train(Category ~.,
                             data = train,
                             method = "rf",
                             trControl = ctrl.rf,
                             tuneGrid = param.grid.rf)
```

```
plot(parameter.search.rf)
```




```
library(randomForest)
set.seed(101)
rf.model <- randomForest(Category ~.,
                         train,
                         mtry = parameter.search.rf$bestTune$mtry,
                         ntree = 10
                         )
```

```
rf.preds <- predict(rf.model, test)
```

Now, let's examine the performance metrics. We observe an accuracy rate of 92.39%. Our Kappa value stands at 0.6483. This indicates that the model we have created performs well. It demonstrates excellent performance in the Blood Donor category but struggles to detect the Suspect Blood Donor category. This is due to the low prevalence of this category, and this issue can be mitigated by adding more data. While the model effectively identifies the Cirrhosis category, it shows moderate performance for the Fibrosis and Hepatitis categories.

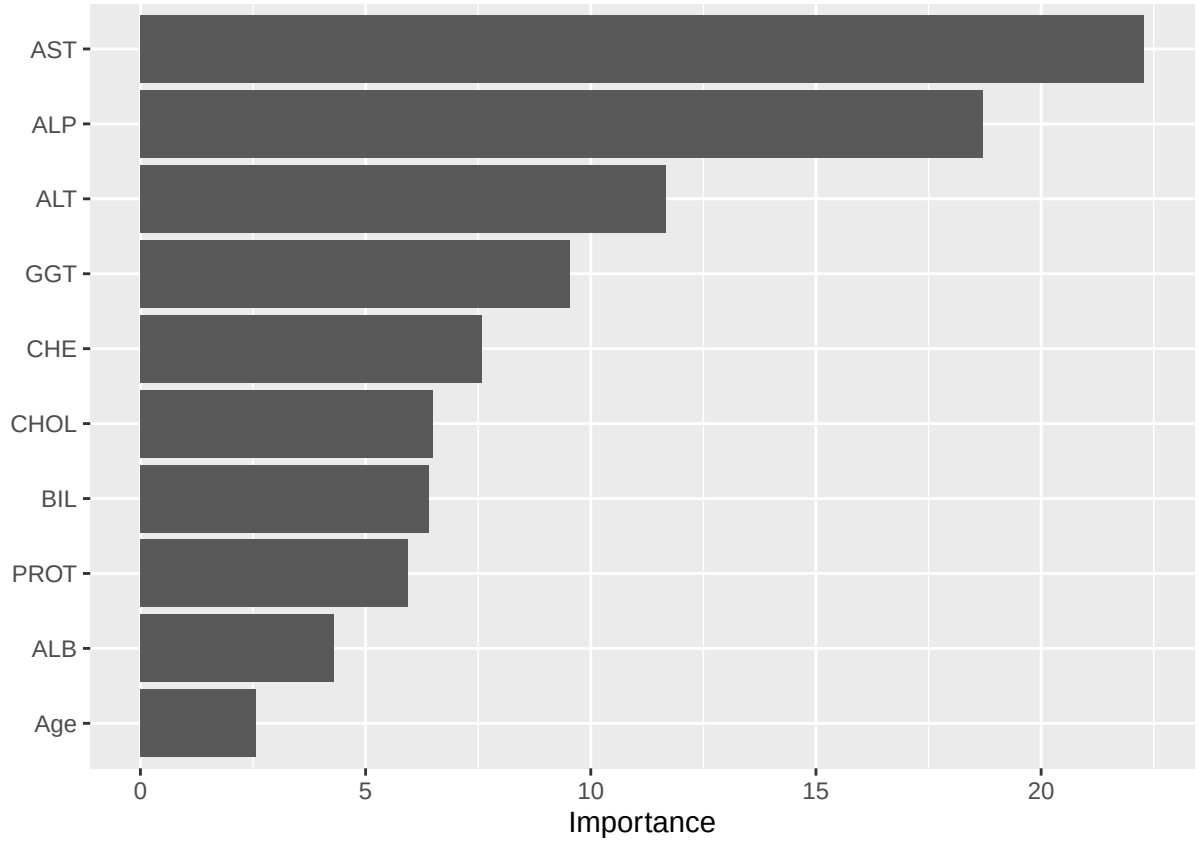
```
confusionMatrix(rf.preds, test$Category)
```

```
## Confusion Matrix and Statistics
##
##                       Reference
```

```
## Prediction            Blood Donor Cirrhosis Fibrosis Hepatitis
##   Blood Donor                 160         2        0         2
##   Cirrhosis                     0         4        0         1
##   Fibrosis                      0         1        3         2
##   Hepatitis                     0         2        3         2
##   suspect Blood Donor           0         0        0         0
##                     Reference
## Prediction            suspect Blood Donor
##   Blood Donor                           1
##   Cirrhosis                             0
##   Fibrosis                              0
##   Hepatitis                             0
##   suspect Blood Donor                   1
##
## Overall Statistics
##
##                Accuracy : 0.9239
##                  95% CI : (0.8756, 0.9578)
##     No Information Rate : 0.8696
##     P-Value [Acc > NIR] : 0.01418
##
##                   Kappa : 0.6483
##
##  Mcnemar's Test P-Value : NA
##
## Statistics by Class:
##
##                      Class: Blood Donor Class: Cirrhosis Class: Fibrosis
## Sensitivity                      1.0000          0.44444         0.50000
## Specificity                      0.7917          0.99429         0.98315
## Pos Pred Value                   0.9697          0.80000         0.50000
## Neg Pred Value                   1.0000          0.97207         0.98315
## Prevalence                       0.8696          0.04891         0.03261
## Detection Rate                   0.8696          0.02174         0.01630
## Detection Prevalence             0.8967          0.02717         0.03261
## Balanced Accuracy                0.8958          0.71937         0.74157
##                      Class: Hepatitis Class: suspect Blood Donor
## Sensitivity                   0.28571                   0.500000
## Specificity                   0.97175                   1.000000
## Pos Pred Value                0.28571                   1.000000
## Neg Pred Value                0.97175                   0.994536
## Prevalence                    0.03804                   0.010870
## Detection Rate                0.01087                   0.005435
## Detection Prevalence          0.03804                   0.005435
## Balanced Accuracy             0.62873                   0.750000
```

When we look at the variable importance plot, we see that the AST value is the most important variable, while the age variable has low importance. A high importance of a variable indicates how much it contributes to and affects the model's prediction. In the model we created, the AST variable has the highest importance. This means that in determining the disease category (Cirrhosis, Fibrosis, etc.), it has a stronger effect than other variables.

```
library(vip)
vip(rf.model)
```



CT MODEL METRICS

Sensitivity (True Positive Rate): High for Blood Donors (0.9938), but significantly lower for Cirrhosis (0.4444) and Hepatitis (0.0000).

Specificity (True Negative Rate): Generally high across all conditions except for Blood Donors (0.7083).

Positive Predictive Value (PPV): Varies widely, with some values not available (NaN) due to zero denominators.

Negative Predictive Value (NPV): Consistently high across all conditions.

Prevalence: Indicates how common each condition is within the sample population.

Detection Rate: Proportional to Sensitivity and Prevalence combined.

Detection Prevalence: Close to or slightly higher than Prevalence rates.

Balanced Accuracy: Moderate to high across most conditions.

RF MODEL METRICS

Sensitivity: Perfect (1.0000) for Blood Donors, but lower for other conditions.

Specificity: High across all conditions.

PPV: Generally good, especially for suspect Blood Donors and Fibrosis.

NPV: High for all conditions.

Prevalence: Same as in the CT model.

Detection Rate: Reflects Sensitivity and Prevalence.

Detection Prevalence: Similar to Prevalence rates.

Balanced Accuracy: Good overall, especially for suspect Blood Donors and Fibrosis.

These metrics help evaluate the models' effectiveness in identifying liver conditions. High sensitivity means good detection of true positives, while high specificity indicates accurate recognition of true negatives. PPV tells us how likely a positive result corresponds to the actual condition, and NPV indicates the likelihood of a negative result being truly negative. These insights are crucial for clinical decision-making.